

Getting Data Out of SAS

It is often necessary to move data out of the SAS system for storage or use in other software. As was the case with moving data into SAS, output can involve SAS datasets, ASCII files, or other data types such as EXCEL spreadsheets.

Permanent SAS Datasets

SAS defines all datasets with two-part names separated by a period. These are in the format: *LIBNAME* . *MEMBER*. The first half, *LIBNAME*, is a library name. It is a link or pointer to a location on the computer. In form, it can take on any alphanumeric id that SAS allows and is defined in a **LIBNAME** statement. For example, if we want to store our datasets at *C:\MYDATA* in a library called *SASCLASS*, then the following statement should be run:

```
LIBNAME SASCLASS 'C:\MYDATA' ;
```

In the **LIBNAME** statement the physical location *C:\MYDATA* should be placed in single quotes. This defines the library. It can hold any number of datasets and its name can change with each SAS session. Also note that more than one library can point to the

same location. By default, if you don't use a library name, SAS puts the data in a library called WORK. It is temporary and will be erased when you exit SAS.

The second half of the name, MEMBER, is what the dataset itself should be called. In our previous examples, we created a dataset named EXP1. Thus to create a permanent dataset at `C:\MYDATA`, we would use something like:

```
DATA SASCLASS.EXP1;  
    INPUT SOIL $ TRT COUNT1 COUNT2;
```

This will create a file in the directory `C:\MYDATA` called `EXP1.*`, where the `*` will be an extension referring to SAS and its version, i.e. `SAS7b`. Anytime you want to use this data, you must call it by its two-part name, i.e.

```
PROC GLM DATA=SASCLASS.EXP1;  
    etc.
```

NOTE: Make sure you have defined the library first with the **LIBNAME** statement (only necessary once per SAS session).

One use of a permanent dataset is to avoid having to import an EXCEL spreadsheet every time you run SAS. To do this, define a permanent library first and then specify it during the import procedure (see handout #4).

Exporting to ASCII (text) Files

To move SAS datasets to ASCII files that can be read by text editors and other software, we use a special call to the data step. When SAS sees this, it runs through the process of the data step without actually creating a dataset. To control the output stage of the data step, we use the inverse of the statements previously covered in data input. Specifically, the statements **FILE** (opposite of **INFILE**) and **PUT** (opposite of **INPUT**) will be used. We also need to use the **SET** statement to pick up some data to work with. For example:

```
DATA _NULL_ ;
    SET EXP1 ;
    FILE 'C:\WORK' ;
    PUT @1 SOIL @5 TRT @10 COUNT1 @15 COUNT2 ;
RUN ;
```

The dataset name **_NULL_** tells SAS not to create a dataset and just run the code given. The **SET** statement picks up our EXP1 dataset+. The **FILE** statement tells SAS where to put the new data file and the **PUT** statement indicates what variables to write out and where in the file to put them. Any syntax that works with **INFILE** or **INPUT** should work with **FILE** and **PUT**, respectively. In this case, column pointers are used to write each variable in the file with 5 columns between values. Care should be taken when outputting data. If you leave too few columns between variables, SAS will overwrite values and run the columns together. In addition, any existing file with the name given in the **FILE** statement will be overwritten by default.

Exporting Other Data Types

When moving SAS datasets to other formats, such as EXCEL, the process will be similar to importing (see handout #4). Choose the EXPORT DATA option from the FILE menu to start the export dialog and then simply follow the screens and options to specify the library and dataset you want to export.